COSMOS: A Conversational Algebraic System

Nagata, Morio; Shibayama, Makoto

数理解析研究所講究録
486 巻 1983 年 124-153

11

COSMOS:
A Conversational Algebraic System

by
Morio Nagata and Makoto Shibayama
(永田守男)　(柴山誠)

ABSTRACT

This report is an introduction to our conversational algebraic system called COSMOS (COnversational Symbolic Manipulation Oriented System). This system is designed as an assistant for engineers and scientists who are not specialists of computers. If the user has a conversation with COSMOS, he or she can make one or a series of following calculations without the writing of programs for his or her problem solving.

(1) Automatic simplification of an expression
(2) Substitution of an expression for a variable
(3) Expansion of a polynomial
(4) Symbolic differentiation
(5) Calculation of symbolic matrices
(6) Solving a univariate polynomial equation
(7) Solving simultaneous linear equations
(8) Solving a univariate linear inequality

COSMOS provides above facilities and interactive facilities by using RLISP and REDUCE 2 on DEC20. The user dose not need to distinguish numerical calculation and symboic calculation. If the system is slightly modified and improved, then it will be a simple personal computing system for engineers and scientists. This report is written as the COSMOS user's manual.

TABLE OF CONTENTS

[I] INTRODUCTION

COSMOS (COnversational Symbolic Manipulation Oriented System) is the conversational algebraic system base on REDUCE 2 [1]. If the user has a conversation with COSMOS, he or she can make one or a series of following calculations without the writing of programs.

(1) Automatic simplification of an expression
(2) Substitution of an expression for a variable
(3) Expansion of a polynomial
(4) Symbolic differentiation
(5) Calculation of symbolic matrices
(6) Solving a univariate polynomial equation
(7) Solving simultaneous linear equations
(8) Solving a univariate linear inequality

COSMOS is designed as an assistant for engineers and scientists who are not specialists of computers. This system assists to solve their problems. In order that they can easily use COSMOS, several facilities are provided. For example, it is some trouble for the user to input the same expressions repeatedly. Thus, this system has back and forth expressions which are differentiated or substituted, and the user can use these expressions without typing (i.e. when the user differentiates an expression "$y=ax^2+bx+c$" with respect to "x" and substitutes "1" for "a" in this

differentiated expression, it saves time and labor for inputing the expression "y'=2ax+b")

COSMOS has been implemented on DEC20 under the TOPS20 operating system by using RLISP and REDUCE 2. However, if the system is slightly modified and improved, then it will be a simple personal computing system for engineers and scientists.

[II] EXAMPLES

2.1 PRELIMINARY

When COSMOS runs, it writes as follows:

```
DO YOU HAVE A CONVERSATION?
PLEASE INPUT Y OR N.
?_
```

In this state, our system waits for the user's input. The _ is a cursor. If the user inputs "N", then the REDUCE system runs. If s/he inputs "Y", then COSMOS writes as follows:

```
WHAT KIND OF CALCULATION DO YOU MAKE?

(1) SIMPLIFICATION OF AN EXPRESSION
(2) SUBSTITUTION
(3) EXPANSION OF A POLYNOMIAL
(4) SYMBOLIC DIFFERENTIATION
(5) CALCULATION OF SYMBOLIC MATRICES
(6) SOLVING A UNIVARIATE POLYNOMIAL EQUATION
(7) SOLVING SIMULTANEOUS LINEAR EQUATIONS
(8) SOLVING A UNIVARIATE LINEAR INEQUALITY

PLEASE INPUT 1 - 8.
?_
```

Here, the user inputs a number which s/he wants to calculate.

According to this menu, we shall show the examples of real conversation records in this section. Every line followed by '?' is the input of the user.

2.2 SIMPLIFICATION OF AN EXPRESSION

If the user selects '1', then the following conversation will continue.

```
WHAT EXPRESSION DO YOU SIMPLIFY?
PLEASE INPUT THE EXPRESSION.
?A+B-2*A+3-2;

B - A + 1

NEXT, WHAT KIND OF CALCULATION DO YOU MAKE?

(1) SIMPLIFICATION OF ANOTHER EXPRESSION
(2) SUBSTITUTION OF THIS EXPRESSION FOR A VARIABLE
(3) SYMBOLIC DIFFERENTIATION OF THIS EXPRESSION
(4) SOLVING THIS EXPRESSION = 0
(5) SOLVING THIS EXPRESSION <> 0
(6) OTHERWISE

PLEASE INPUT 1 - 6.
?1

WHAT EXPRESSION DO YOU SIMPLIFY?
PLEASE INPUT THE EXPRESSION.
?2*X+3*X**2=X+3;

   2
3*X  + X - 3 = 0

       :
       :
       :
```

2.3 SUBSTITUTION

If the user selects '2' of the menu shown in 2.1, then the following conversation will continue.

```
PLEASE INPUT THE EXPRESSION TO BE SUBSTITUTED.
?A*X**2+B*X+C;
```

```
IF YOU WANT TO SUBSTITUTE 'A' FOR 'B',
THEN TYPE 'A=B;'.
?X=1;

B + A + C


NEXT, WHAT KIND OF CALCULATION DO YOU MAKE?

(1) ANOTHER SUBSTITUTION
(2) PROCESSING OF THIS EXPRESSION
(3) PROCESSING OF THE EXPRESSION BEFORE SUBSTITUTED
(4) OTHERWISE

PLEASE INPUT 1 - 4.
?_
```

:
:
:

2.4 EXPANSION OF A POLYNOMIAL.

When the user would like to expand a polynomial, the following conversation will continue.

```
PLEASE INPUT THE EXPRESSION TO BE EXPANDED.
?(A+B+C)**2;

 2                    2           2
B  + 2*B*C +2*B*A + C  + 2*C*A + A


NEXT, WHAT KIND OF CALCULATION DO YOU MAKE?

(1) EXPANSION OF ANOTHER EXPRESSION
(2) SUBSTITUTION
(3) DIFFERENTIATION OF THIS EXPRESSION
(4) SOLVING THIS EXPRESSION = 0
(5) SOLVING THIS EXPRESSION <> 0
(6) OTHERWISE

PLEASE INPUT 1 - 6.
?_
```

:
:
:

2.5 SYMBOLIC DIFFERENTIATION

Now, let us consider that the user would like to differentiate an expression with respect to a variable.

```
WHAT EXPRESSION DO YOU DIFFERENTIATE?
PLEASE INPUT THE EXPRESSION.
?A*X**3+B*X**2+C+1;

PLEASE SPECIFY THE VARIABLE.
?X

WHAT TIMES DO YOU DIFFERENTIATE THE EXPRESSION?
PLEASE INPUT THE NUMBER.
?2


6*X*A + 2*B

NEXT, WHAT DO YOU CALCULATE?

(1) DIFFERENTIATION OF ANOTHER EXPRESSION
(2) PROCESSING OF THIS EXPRESSION
(3) PROCESSING OF THE EXPRESSION BEFORE DIFFERENTIATED
(4) OTHERWISE

PLEASE INPUT 1 - 4.
?_
```

:
:
:

2.6 CALCULATION OF SYMBOLIC MATRICES

If the user would like to make calculations of matrices, s/he will have the following conversation.

```
WHAT MATRIX NAMES DO YOU USE?
PLEASE INPUT THE NAME LIST OF MATRICES.
?A,B,C;

PLEASE INPUT MATRIX A.
```

```
1-ST ROW
?1,2,3;
2-ND ROW
?1,0,1;
3-RD ROW
?2,3,0;
4-TH ROW
?;

NEXT, PLEASE INPUT MATRIX B.

1-ST ROW
?1,0,0;
2-ND ROW
?0,1,0;
3-RD ROW
?0,0,1;
4-TH ROW
?;

NEXT, PLEASE INPUT MATRIX C.

1-ST ROW
?X+Y,1,1;
2-ND ROW
?X,0,Y;
3-RD ROW
?1,0,X-Y;
4-TH ROW
?;


WHAT KIND OF CALCULATION DO YOU MAKE?
 INVERSE A => A**(-1), TRANSPOSE A => TP(A)
 TRACE A => TRACE(A), DETERMINANT A => DET(A)
 EIGENVALUE A => EIG(A)

PLEASE INPUT THE EXPRESSION.
?A+B-C;


MAT(1,1) := - X - Y + 2

MAT(1,2) := 1

MAT(1,3) := 2

MAT(2,1) := - X + 1

MAT(2,2) := 1

MAT(2,3) := - X + 1

MAT(3,1) := 1
```

```
MAT(3,2) := 3

MAT(3,3) := - X + Y + 1



NEXT, WHAT KIND OF CALCULATION DO YOU MAKE?

(1) ANOTHER MATRIX CALCULATION
(2) OTHERWISE

PLEASE INPUT 1 OR 2.
?1

DO YOU ADD NEW MATRICES?
PLEASE TYPE Y OR N.
?N

PLEASE INPUT THE EXPRESSION.
?D=TP(A);

D(1,1) := 1

D(1,2) := 1

D(1,3) := 2

D(2,1) := 2

D(2,3) := 0

D(3,1) := 3

D(3,2) := 1

D(3,3) := 0

     :
     :
     :
```

2.7 SOLVING A UNIVARIATE POLYNOMIAL EQUATION

Now let us consider that the user would like to solve a univariate polynomial equation.

```
WHAT EQUATION DO YOU SOLVE?
PLEASE INPUT THE EQUATION.
?X**3-6*X**2+11*X-6=0;

PLEASE INPUT THE VARIABLE TO BE SOLVED.
?X

X(1) := 2

X(2) := 3

X(3) := 1

NEXT, WHAT KIND OF CALCULATION DO YOU MAKE?

(1) SOLVE ANOTHER UNIVARIATE POLYNOMIAL EQUATION
(2) SOLVE THIS EQUATION WITH RESPECT TO ANOTHER VARIABLE
(3) PROCESSING OF THE GIVEN EQUATION
(4) OTHERWISE

PLEASE INPUT 1 - 4.
?_
```

:
:
:

2.8 SOLVING SIMULTANEOUS LINEAR EQUATIONS

In the case of solving simultaneous linear equations, the following conversation will continue.

```
WHAT LINEAR EQUATION DO YOU SOLVE?
PLEASE INPUT THE EQUATIONS TO BE SOLVED,
AND TYPE ';' AT THE LAST OF THEM.
?X+2*Y+Z=3;
?X+3*Y=1;
?Z-Y=2;
?;

PLEASE INPUT THE LIST OF VARIAVLES TO BE SOLVED
(E.G. X,Y,Z;).
?X,Y,Z;

X := X

Y := ( - X + 1)/3
```

```
Z := ( - X + 7)/3


NEXT, WHAT CALCULATION DO YOU MAKE?

(1) SOLVE OTHER LINEAR EQUATIONS
(2) PROCESSING OF THE GIVEN SET OF EQUATIONS
(3) OTHERWISE

PLEASE INPUT 1 - 3.
?2


WHAT KIND OF CALCULATION DO YOU MAKE?

(1) SUBSTITUTION
(2) SOLVE THIS SET OF EQUATIONS WITH RESPECT
    TO ANOTHER SET OF VARIABLES
(3) ADD ANOTHER EQUATION AND SOLVE
(4) DELETE AN EQUATION AND SOLVE
(5) PROCESSING OF ONE EQUATION IN THIS SET

PLEASE INPUT 1 - 5.
?1

(1) X + 2*Y + Z = 3

(2) X + 3*Y = 1

(3) Z - Y = 2


WHAT EXPRESSIONS DO YOU WORK?
PLEASE INPUT 1 - 3, ALL
EXAMPLE '1;', '1,3;', 'ALL;'.
?3;

WHAT KIND OF SUBSTITUTION DO YOU MAKE?
IF YOU WANT TO SUBSTITUTE 'A' FOR 'B', AND 'C' FOR 'D',
THEN TYPE 'A=B,C=D;'.
?Y=-Y;

(1) X + 2*Y + Z = 3

(2) X + 3*Y = 1

(3) Z + Y - 2 = 0

MAY I SOLVE THESE EQUATIONS?
PLEASE TYPE Y OR N.
?Y

PLEASE INPUT THE LIST OF VARIABLES TO BE SOLVED
(E.G. X,Y,Z;).
?X,Y,Z;
```

```
X := 1

Y := 0

Z := 2

    :
    :
    :
```

2.9 SOLVING A UNIVARIATE LINEAR INEQUALITY

When the user would like to solve a univariate linear inequality, s/he will have the following conversation.

```
WHAT INEQUALITY DO YOU SOLVE?
HERE, USE '>=' INSTEAD OF '> OR ='
          '<='             '< OR ='

PLEASE INPUT AN INEQUALITY.
?A*X+B>0;

PLEASE INPUT THE VARIABLE TO BE SOLVED.
?X

A > 0  =>  X > ( - B)/A

A < 0  =>  X < ( - B)/A

A = 0  =>  B > 0 : TRUE

           B <= 0 : FALSE


NEXT, WHAT CALCULATION DO YOU MAKE?

(1) SOLVE ANOTHER UNIVARIATE LINEAR INEQUALITY
(2) SOLVE THIS INEQUALITY WITH RESPECT TO ANOTHER VARIABLE
(3) PROCESSING OF THE GIVEN INEQUALITY
(4) OTHERWISE

PLEASE INPUT 1 - 4.
?_
```

```
    :
    :
    :
```

[III] REFERENCE MANUAL

3.1 STRUCTURE OF EXPRESSIONS

3.1.1 The COSMOS Standard Character Set

The basic characters which are used to build up COSMOS symbols are the following:

(1) The 28 upper case letters A through Z

(2) The 10 decimal digits 0 through 9

(3) The special characters ( ) + - * / , . ; < > =

3.1.2 Numbers

A number in COSMOS may be of two types - integer and fraction. Integers consist of a signed or unsigned sequence of decimal digits written without a decimal point.

e.g. -1, 415, +3163

Fractions consist of <integer>/<unsigned integer>.

e.g. 1/2, -10/21, +3/7

3.1.3 Idetifiers

Identifiers in COSMOS consist of 1 to 24 alphanumeric characters (i.e. upper case alphabetic letters or decimal

digits) the first of which must be alphabetic.

e.g. A, ABC, A1B2C3

Identifiers are used as variables, symbolic constants and operators (e.g. SIN, COS, TAN). But reserved words in COSMOS may not be used as identifiers, and an identifiers may not extend over a line of text.

3.1.4 Variables

Variables are a particular type of identifier.

Reserved Variables

Several variables in COSMOS have a paticular value which can not easily be changed by the user. These variables are as follows:

I : square root of -1. I**2 = -1

E : base of natural logarithm. LOG(E) = 1

3.1.5 Operators

Operators in COSMOS are the same as ones in FORTRAN.

A + B --> A+B

A - B --> A-B

A × B --> A*B

A ÷ B --> A/B

A^B --> A**B

-A --> -A

and operator procedence is

unary > ** > * / > + -

and it is possible to use parentheses in an expression.

e.g. (1+2)*(3-2**3)+1

3.1.6 Logical Operators

Logical operators in COSMOS are as follows

A = B --> A=B

A > B --> A>B

A ≧ B --> A>=B

A < B --> A<B

A ≦ B --> A<=B

3.1.7 Reserved Words

Certain words are reserved in COSMOS. The following words may only be used in the manner decribed in this

manual.

BEGIN	DIFFERENCE	DO
ELSE	END	FOR
FUNCTION	GO	GOTO
IF	LAMBDA	MINUS
NIL	PLUS	PRODUCT
QUOTIENT	RETURN	STEP
SUM	THEN	TIMES
TO	UNTIL	WHILE

3.2 FUNCTIONS

3.2.1 DF(Differentiation)

COSMOS has a function of differentiation, however, the operator DF is used to represent partial differentiation with respect to one variable. The first argument is the scalar expression to be differentiated. The second argument specifies the differentiation variable, and the third argument specifies the number of times. They are applied by the following syntax:

DF(<expression>,<variable>,<number>)

The <number> may be omitted if it is 1.

e.g. DF(X**2,X,2) = 2

DF(SIN(X),X) = COS(X)

3.2.2 SIN, COS, TAN, COSEC, SEC, COTAN

These elementary functions are included in COSMOS. Automatic simplification rules of these functions are as follows:

```
   SIN(-X) = - SIN(X)
   COS(-X) = COS(X)
   TAN(-X) = - TAN(X)
 COSEC(-X) = - COSEC(X)
   SEC(-X) = SEC(X)
 COTAN(-X) = - COTAN(X)
    SIN(0) = 0
    COS(0) = 1
    TAN(0) = 0
    SEC(0) = 1
```

3.2.3 ARCSIN, ARCCOS, ARCTAN, LOG

These elementaly functions are included in the system.

```
ARCSIN(-X) = - ARCSIN(X)
ARCTAN(-X) = - ARCTAN(X)
```

ARCSIN(0) = 0

ARCCOS(1) = 0

ARCTAN(0) = 0

LOG(1) = 0

LOG(E) = 1

3.2.4 SQRT(square root), CBRT(cubic root)

These elementary functions are included in the system with fundamentals.

ex. SQRT(0) = 0

SQRT(1) = 1

SQRT(4) = 2

SQRT(8) = 2*SQRT(2)

SQRT(9) = 3

SQRT(12) = 2*SQRT(3)

:
:
:

3.2.5 Operators with Matrix Arguments

Four additional operators are useful in matrix calculations, namely DET, TP, TRACE and EIG defined as follows:

determinant --> DET(A)

transpose --> TP(A)

trace --> TRACE(A)

eigenvalue --> EIG(A)

inverse --> 1/A or A**(-1)

and A/B is interpreted as A*B**(-1).

ex. If both X and Y are matrix, then the following are matrix expressions.

```
X
X + Y**2
X - TP(Y) + X/Y
DET(X)
```

3.3 CONVERSATION

3.3.1 Preliminary

It takes conversation with COSMOS for the user to input an answer to questions given by COSMOS, expressions and so on. The user can input anything followed by the symbol

?_,

where _ is a cursor.

3.3.2 An Answer to Questions (Menu)

When the user wants to input an answer to the question givend by COSMOS, s/he inputs the number without ";"(semicolon).

```
(1) SIMPLIFICATION OF AN EXPRESSION
(2) SUBSTITUTION
(3) EXPANSION OF A POLYNOMIAL
(4) SYMBOLIC DIFFERENTIATION
(5) CALCULATION OF SYMBOLIC MATRICES
(6) SOLVING A UNIVARIATE POLYNOMIAL EQUATION
(7) SOLVING SIMULTANEOUS LINEAR EQUATIONS
(8) SOLVING A UNIVARIATE LINEAR INEQUALITY

PLEASE INPUT 1 - 8.
?3


PLEASE INPUT 1 - 8.
?3;
is not allowed
```

3.3.3 Expressions

When the user inputs an expression to COSMOS, s/he writes an expression like FORTRAN with ";". If the user misses a semicolon, it comes "?" again. Thus, if s/he realy ends the expression, then s/he inputs only semicolon.

```
e.g.  ?X
      ?+
      ?Y
      ?;

      is same as
      ?X + Y;
```

3.3.4 Yes-No

If the user wants to input YES or NO, s/he inputs only

Y(yes) or N(no) without the semicolon.

ex.
PLEASE INPUT Y OR N.
?Y

3.3.5 Otherwise

atom -- without semicolon

WHAT TIMES DO YOU DIFFERENTIATE THE EXPRESSION?
PLEASE INPUT THE NUMBER.
?2

list -- with semicolon

ex.
PLEASE INPUT MATRIX A.

1-ST ROW
?1,2,3;
2-ND ROW
?X+Y,Z,X*Y+Z;

:
:
:

[IV] DESIGN OF THE NEW SYSTEM

Our early experience with COSMOS has shown the usefulness of this system for the solving process of simple problems. However, we have recognized that we should improve the following points of COSMOS by the experience.

1) Menus, input and output expressions are displayed in disorder.
2) Many menu statements written in English are displayed, thus we can not understand them at a glance.
3) Many facilities of COSMOS are implemented by using those of REDUCE 2, so COSMOS has less portability.
4) It is desired that symbilic and numerical computation are effectively combined.

Therfore, in order to improve these points, we have designed a new version of COSMOS. In our new system, constants, functions and expressions are kept in the data base, and these are used if necessary. An overview of this system is given in Figure 1, and all parts of the new system are explained briefly.

1) Symbolic manipulation part

This is the main part of the new system. This part manipulates symbolic expressions. It provides facilities only for simultaneous linear equations now. This manipulation is both a general symbilic manipulation and a pre-manipulation which is easy to be analyzed numerically.

2) Numerical anarysis part

We can obtain the numerical solution of processed equations if necessary.

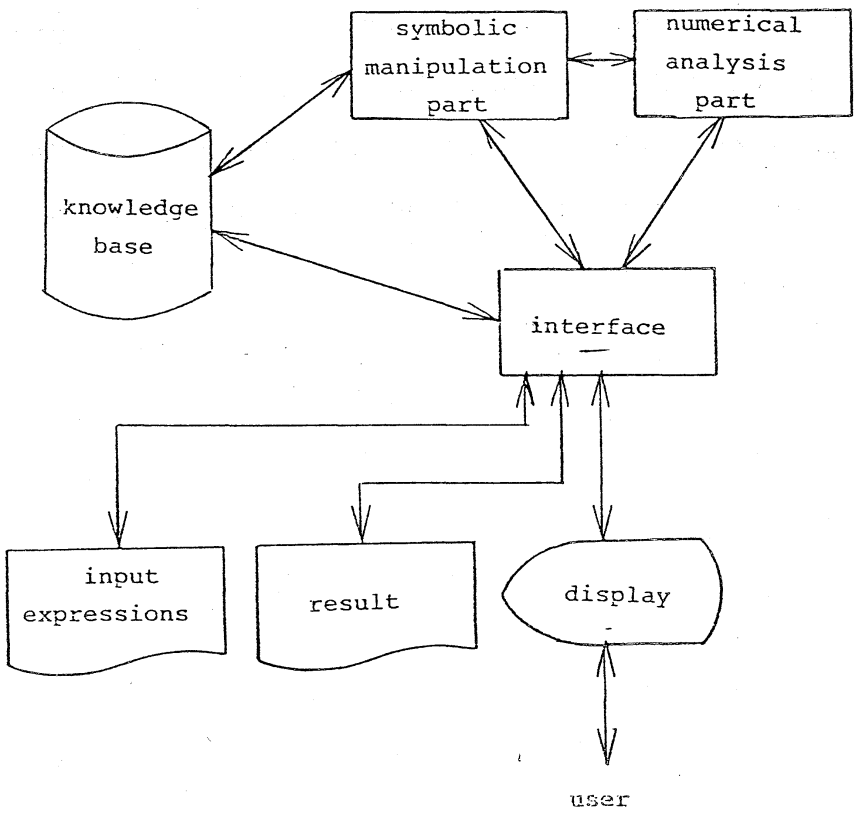

FIGURE 1.

3) Knowledge base

It has knowledge of matrix. It also keeps user's constants, functions and expressions which are alredy given by the user. These are refered by the system in the problem solving process.

4) Interface

The new system provides a man-machine interface which helps us to use COSMOS. Menus, inputs and outputs are displayed in separated areas. Menu statements of this interface are shown in Japanese sentences including Chinese characters.

[V] CONCLUSIONS

COSMOS is now running on DEC2020 installed in Keio Institute of Information Science. We have not gained much experience in using this system. However, we have ascertained the effectiveness of COSMOS for certain types of problem solving. On the other hand, we have found that the following problems should be solved.

(1) Addition of facilities (Solving a differential equation, Graphics, etc.)
(2) Efficient algorithms
(3) Flexibility (Open-ended system)
(4) Close man-machine interaction
(5) Implementation on the personal computer

If these problems are solved, COSMOS will be a practical personal computing system for engineers and scientists.

ACKNOWLEDGEMENTS

We would like to express our gratitude to Dr. Anthony C. Hearn of Rand Corporation, Dr. Tateaki Sasaki and Mr. Nobuyuki Inada of Institute for Physical and Chemical Reserch for their kind advices on using REDUCE 2. Authors also thank research staffs of Keio Institute of Information Science for their permission of our use of DEC20. We specially thank to Prof. Shoji Ura and Dr. Zentaro Nakamura for their useful suggestions.

REFERENCE

1. Hearn, A.C.: REDUCE 2 USER'S MANUAL
Univ. of Utah (1973)